AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing Of Claims:

1-12. (Canceled)

13. (New) A method for transcribing at least one data record of an external data source to a processor unit, comprising:

transmitting the at least one data record from the external data source together with additional information to a buffer memory of the processor unit, the additional information including an identifier assigned individually to the processor unit;

performing, in the processor unit, a check of an admissibility of a use of the at least one data record on the basis of the additional information;

generating a blocking signal when the check indicates that the use of the at least one data record is not allowed;

deleting the at least one data record from the buffer memory; and

generating an enable signal when the use of the at least one data record is allowed.

14. (New) The method as recited in Claim 13, wherein the identifier is valid only once for checking the at least one data record that has been transmitted and stored in the buffer memory.

15. (New) The method as recited in Claim 13, wherein when the enable signal has been generated, the at least one data record is transmitted from the buffer memory to a functional memory from which the at least one data record may be read.

16. (New) The method as recited in Claim 15, further comprising:

storing the identifier and the at least one data record in the functional memory; and

checking the identifier when calling up the at least one data record from the functional memory.

17. (New) The method as recited in Claim 13, further comprising:

storing a list of code words in a code word memory of the processor unit;

comparing the identifier with at least one of the code words; and

determining a presence of validity when the at least one of the code words and the identifier match.

18. (New) The method as recited in Claim 13, further comprising:

storing a list of code words in a code word memory of the processor unit;

comparing the identifier with at least one of the code words; and

determining a presence of validity when the at least one of the code words and the identifier are identical.

19. (New) The method as recited in Claim 17, further comprising:

storing a counter content of a counter, wherein the counter points to one of the code words of the code word memory; and

incrementing the counter content of the counter before each check of the identifier of the at least one data record stored in the buffer memory.

20. (New) The method as recited in Claim 13, wherein:

the processor unit is identifiable by an identification sequence, and

the identification sequence is part of the additional information and

is used in the check of the at least one data record.

21. (New) The method as recited in Claim 13, further comprising:

storing valid identifiers for the processor unit in an identifier server.

22. (New) The method as recited in Claim 13, further comprising:

retrievably storing identifiers in a code word server for a plurality of counter contents of a counter, the identifiers being allocatable to the processor unit via an identification sequence.

23. (New) A processor unit, comprising:

a buffer memory;

a rewritable functional memory that is accessible during an operation of the processor unit, the buffer memory and the rewritable function memory being capable of storing at least one data record;

an interface for importing the at least one data record and additional information into the buffer memory; and

a check unit for checking a validity of the at least one data.

24. (New) The processor unit as recited in Claim 23, further comprising:

a read-only code word memory for storing code words; and

a counter including an incrementable counter content that points to one of the code words, wherein the processor unit is individualized via an identification sequence.

25. (New) The processor unit as recited in Claim 23, wherein the processor unit is a control unit of a motor vehicle.